

Upgrading NASA/DOSE Laser Ranging System Control Computers

by

Randall L. Ricklefs

McDonald Observatory, University of Texas, Austin, Texas 78712

Jack Cheek

Hughes-STX, Lanham, MD 20706

Paul J. Seery, Kenneth Emenheiser, William P. Hanrahan
Bendix Field Engineering Corp., Seabrook, MD 20706

Jan F. McGarry

NASA Goddard Space Flight Center, Greenbelt, MD 20771,

Laser ranging systems now managed by the NASA Dynamics of the Solid Earth (DOSE) and operated by the Bendix Field Engineering Corporation, the University of Hawaii, and the University of Texas, have produced a wealth of inter-disciplinary scientific data over the last three decades. Despite upgrades to the most of the ranging station subsystems, the control computers remain a mix of 1970s-vintage minicomputers. These encompass a wide range of vendors, operating systems, and languages, making hardware and software support increasingly difficult. Current technology allows replacement of controller computers at a relatively low cost while maintaining excellent processing power and a friendly operating environment. The new controller systems are now being designed using IBM-PC-compatible 80486-based microcomputers, a real-time Unix operating system (LynxOS), and X-windows/Motif graphical user interface. Along with this, a flexible hardware design using CAMAC, GPIB, and serial interfaces has been chosen. This design supports minimizing short and long term costs by relying on proven standards for both hardware and software components. Currently, the project is in the design and prototyping stage with the first systems targeted for production in mid-1993.

Introduction

In the more than two decades of laser ranging, the observational accuracy of the NASA-funded laser stations has dropped from tens of centimeters to a centimeter or better. To achieve this, most parts of the ranging system have been upgraded a number of times. It has long been recognized that the computer subsystems are antiquated and unable to keep up with the requirements for ever-more-precise range measurements and an ever-growing list of targets. In the last year, a plan of action has been formulated to upgrade these systems in such a way as to minimize both short and long term cost by making extensive use of standard hardware and system software. The stations involved are the DOSE-operated GSFC 1.2m telescope, the Bendix Field Engineering Corporation-operated MOBLAS 4-8 and TLRs 2-4, the University of Hawaii-operated HOLLAS station, and the University of Texas-operated MLRS

History

The controller computers on the DOSE network laser stations are late-1970s minicomputers produced by vendors such as Modcomp, Data General, and DEC. While they were real workhorses in their day, they are now eclipsed by faster, more expandable microcomputers an order of magnitude less expensive. Much of the code on these machines is written in archaic versions of FORTRAN which fail to meet even the FORTRAN 66 standards. A great deal of code is also written in the native assembly language of these machines. Many approximation and shortcuts were required to fit complex algorithms into these low-capacity computers. In some cases the operating systems are 'homebrew', while in other cases extensive changes have been made to no-longer-supported proprietary operating systems. If key personnel were to leave these organizations, the computer systems would be virtually unsupportable.

In addressing these issues, the CDP SLR Computer Committee decided in 1988 to off-load the data formatting, communications, and prediction functions of the existing controlling computers to dedicated Hewlett-Packard Unix workstations. These workstations were to also take on the tasks of data filtering and normalpointing. It was recognized at that time that replacing the controllers was a necessary task, but was too complex for the timetable necessary to support the burgeoning list of satellite targets. Figure 1 is a schematic of the current dual-computer system at the stations, with a serial line linking the computers. Since the process of implementing the new computers in the network laser stations is nearly complete, work has begun on the task of replacing the controller computers, a task made easier by the off-loading of many tasks to the workstations.

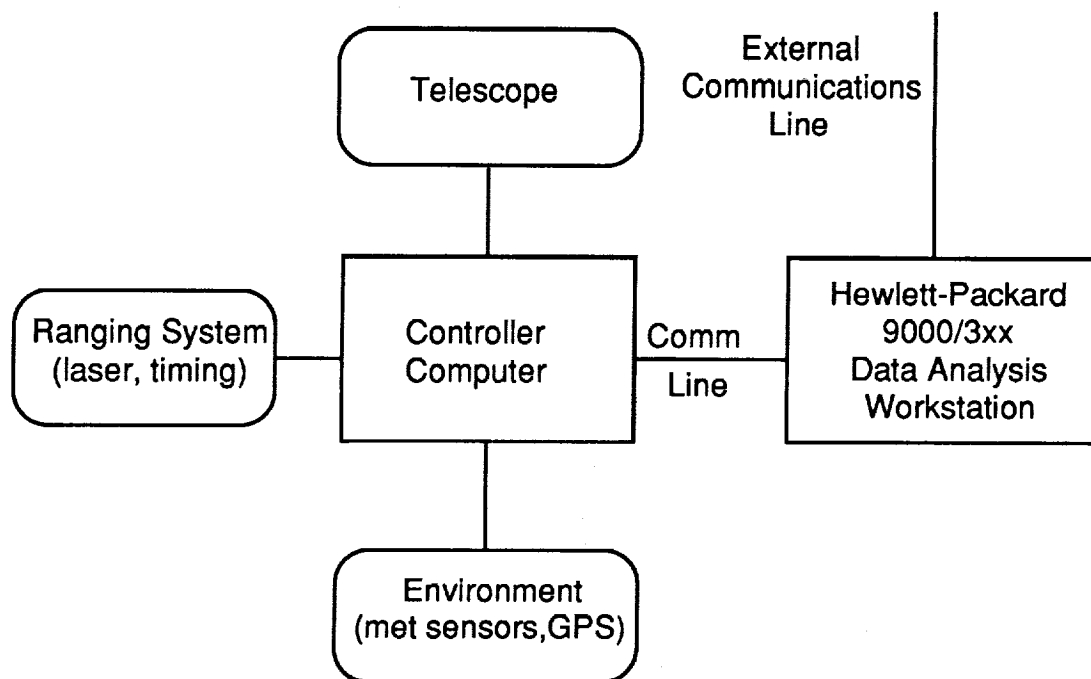


Figure 1: On-site Computer Configuration

Hardware Environment

The new controller system must accommodate many of the vagaries of 7 or so laser systems' designs and philosophies and must be flexible enough to permit expansion of these systems in various ways. For instance, most of the Bendix-operated systems and the University of Hawaii system fire their lasers at 5 or 8 Hz, use external pre- and post-calibrations, and use Hewlett-Packard 5370 counters as time interval units. MLRS on the other hand, fires at 10 Hz, uses internal calibrations and uses a TD811 time digitizer as the finest part of its 3-tiered epoch timing system. The MOBLAS and transportable systems are limited to satellites up to the orbits of the Etalons. The GSFC 1.2m research system accommodates many experimenters, and ranges to aircraft as well as to satellites. MLRS ranges all currently targeted CDP satellites as well as MP-2 (in geostationary orbit) and reflectors on the moon. Some systems use CAMAC interfaces, while others use a proprietary bus interface standard. Some incorporate GP-IB (IEEE-488), and most use serial ports for various devices.

To replace the current controller computers, a new computer must be identified that can talk with each of the existing interfaces and that is likely to be supportable for the foreseeable future. It must support one or more real-time operating systems, and must be fast enough to meet all near term needs. Proprietary architecture and all the danger that implies must be avoided. In addition, the new system must be inexpensive. The obvious choice is the ubiquitous IBM-PC-compatible microcomputer. Boards exist that interface the PC to almost any variety of hardware; the upgrade path to an even faster machine than the prototypical 80486 33 MHz system is expected to be virtually painless; it is relatively inexpensive; and there are a number of reasonable real-time operating systems available.

A prototype systems design was developed which includes an IBM-PC clone containing a 33 Mhz Intel 80486, 16 Mb of RAM, a 330 Mb SCSI hard disk drive, 3 1/2" and 5 1/4" floppy disk drives, a 16 bit ethernet card, high resolution (1024 x 768) video display and a 16" monitor. Interface cards for CAMAC, GP-IB, and a serial mux board were also included. To save cost and space, the preliminary configuration will have the PC use both the printer and erasable optical disk on the HP workstation across the ethernet network.

To standardize the system interfaces and to limit dependence on specific vendors, it was decided to use the CAMAC, IEEE-488 (GP-IB), and RS-232 serial interfaces rather than dedicating a bus slot to virtually each device in a ranging station. Along these lines, the MOBLAS systems' obsolete IACC interface will be replaced by CAMAC, thereby removing an important source of maintenance difficulties. The serial ports are probably the easiest of all of these interfaces to accommodate with the PCs; they will be used to bring in environmental sensor data.

Operating System

Each laser system involved in this project has evolved along a different path dictated by different needs. One overarching need now, however, is to create as much commonality as possible and to program for a software environment that will still be viable in 10 years. There were several options available when the search was made for operating systems, including writing it ourselves. Several of the current stations run home-built operating systems. The difficulties with these have been that they are hard to write and maintain. These systems were written because existing operating systems at the time did not offer the combination of speed, performance, and code size that was necessary to make the most

effective use of the hardware in a real-time environment. With fast and (relatively) inexpensive hardware, there are no such needs today.

In the search for an operating system, we considered DOS and DOS add-ons, various versions of Unix, and several proprietary operating systems, such as Digital Equipment Corporation's VMS and the Apple Macintosh Finder. The proprietary operating systems were rejected immediately, due to our need for vendor-independence. DOS and the Digital Research clone DR-DOS are neither multitasking, multiuser, nor realtime. DOS is also targeted to Intel 80x86 microprocessors, violating our requirement for vendor independence. Although products such as Microsoft Windows and DeskView offer some degree of multitasking, they do not claim to be either pre-emptive or real-time.

Unix was chosen because it has become a commonly available operating system with a rich software development environment. It does provide multi-user and multi-tasking capabilities and is available on many types of processors. Although there are several 'flavors' of Unix in the marketplace, the trend towards standards compliance allows well written programs to be transferred among Unix systems. The Hewlett-Packard workstations already available at the network laser stations run in a Unix environment. Standard Unix is not a real-time operating system. However, there are Unix clones and derivatives which do support realtime, pre-emptive multi-tasking and are designed for realtime acquisition and control applications like laser ranging.

A number of Unix and Unix-like systems were considered. These include:	
Coherent (Mark Williams)	QNX (Quantum Systems)
REAL/IX (Modcomp)	Unix / Xenix (SCO)
VRTX (RadiSys Corp)	VxWorks (Wind River Systems)

The above were eliminated due to one or more of the following reasons. The operating system

- requires proprietary hardware
- did not mention POSIX or real-time standards compliance
- lacks one or more of the required compilers (FORTRAN, C, or assembler)
- lacks network capabilities
- lacks graphics capabilities

The chosen operating system is LynxOS by Lynx Real-Time Systems, Inc. LynxOS is a Unix-clone (no AT&T code) which complies with the POSIX standard as well as IEEE 1003.4 and 1003.4a draft real-time standards. It also follows the 80386 binary file standards and runs on IBM-PC compatible computers. LynxOS is also available for Motorola 680x0 systems and is being ported to the SPARC engine. C compilers (including one by GNU) are supplied with LynxOS, while FORTRAN is available from a third party. TCP/IP, NFS, X windows and OSF Motif are also available from Lynx. It should be mentioned for completeness that soon after LynxOS was chosen, Venix (Venturcom) was discovered. It uses AT&T system V code modified for real-time applications and also run on IBM-PC compatibles. It appears to be a viable alternative choice for a real-time operating system.

User Interface and Networks

Several requirements for the user interface were advanced during planning. These included the need for a user-friendly interface, more graphics and less text, remote monitoring and control, automation of many hardware settings and seamless integration

with the HP workstation. The emerging standard for network-based, vendor-independent graphical user interface is the X windows system. Since X windows specifies no policy or style, there is no consistency required of applications. For this reason the Open Systems Foundation developed the Motif graphical user interface (GUI) on top of X windows, which does have a particular "look and feel". Open Look, which is another GUI supported by AT&T and Sun, does not seem to be as widely available at this time. Using the constraints of the GUI, user friendly applications can be produced that have many of the features of the familiar Apple Macintosh and Microsoft Windows applications, such as menu bars, pop-up windows, and freedom from a command line interface. Also, since X windows is network based, one can use the on-site software from virtually anywhere in the world via a network connection. Much of the experience gained and code developed in implementing such an interface for the on-site Hewlett-Packard workstations for the NASA-funded stations will be of great value in constructing the controller system.

Connecting the controller computer and on-site data-reduction workstation via ethernet and TCP/IP opens a world of data and control-sharing possibilities. Data can be recorded on the controller's disk drive in a directory which is mounted on the workstation via NFS (Network File System). The workstation can then have instant access to the data in real time. Sockets, RPC (Remote Program Calls), and remsh (remote shell) can all serve as tools for creating a seamless dual computer system.

Preliminary Design

In addition to agreeing on prototype hardware and software platforms, the GSFC 1.2m telescope, Bendix, and the University of Texas groups have all acquired prototype systems. Work is proceeding on the design of the user interface and underlying functions. Device drivers for two critical pieces of equipment - the CAMAC and the timing board (used to provide the computer with 20 Hz interrupts from the station clock) - have been written. The monitor program, which runs at all times and maintains information such as time, weather, and telescope status, and the star calibration program will be the first software package to be implemented, as they fulfill needs common to all stations. Work is also proceeding on replacement of the MOBLAS IACC interface with CAMAC. The goal is to put the first three upgraded systems (GSFC 1.2m, MOBLAS 7, and MLRS) into production with the new controller in mid to late 1993.

A preliminary version of the user interface is shown in figure 2. This shows the common display of station parameters maintained by the monitor program such as date, time and telescope position. Also evident is the pull-down menu and display of graphs and parameters unique to the star calibration program.

Conclusion

The overdue upgrade of NASA ranging station control computers is finally coming about and is making use of the latest technology to produce a flexible, extensible, and maintainable computer platform. The upgrade is based on standard hardware and software components to lessen vendor dependence and enhance software portability. The prototype computer system has been designed and acquired by the principals in this upgrade, who are in the midst of further software design and low level software implementation.

File	Edit	Tracking	Diagnostics	Analysis	Communications	Schedule	Help
<div> Delete... View/Print... Backup... Restore... Disk Maintenance... Set Date/Time Reboot System Shutdown System Print... Exit </div>							
992 (182)				STANDBY			
0.0000				Barp 1010.3			
.1000				Temp 10.2			
				Humid 40.3			
Meta+F4							

File	Edit	Tracking	Diagnostics	Analysis	Communications	Schedule	Help
<div> Get IRVs Get EOP Connect to Modem Connect to Network </div>							
Wed Jul 1 09:47:52 1992 (182)				STANDBY			
Mount 123.1230 20.0000				Barp 1010.3			
Biases 2.3000 1.1000				Temp 10.2			
				Humid 40.3			

Figure 2 - Preliminary Monitor program user interface